

(19) 世界知的所有権機関  
国際事務局



(43) 国際公開日  
2003 年 7 月 10 日 (10.07.2003)

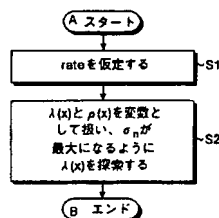
PCT

(10) 国際公開番号  
WO 03/056705 A1

- (51) 国際特許分類: H03M 13/19, H04L 1/00 区丸の内二丁目2番3号三菱電機株式会社内 Tokyo (JP).
- (21) 国際出願番号: PCT/JP02/13518
- (22) 国際出願日: 2002 年 12 月 25 日 (25.12.2002) (74) 代理人: 酒井 宏明 (SAKAI, Hiroaki); 〒100-0013 東京都千代田区霞が関三丁目2番6号東京倶楽部ビルディング Tokyo (JP).
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語 (81) 指定国 (国内): CN, US.
- (30) 優先権データ:  
特願 2001-397922  
2001 年 12 月 27 日 (27.12.2001) JP (84) 指定国 (広域): ヨーロッパ特許 (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, SI, SK, TR).
- (71) 出願人 (米国を除く全ての指定国について): 三菱電機株式会社 (MITSUBISHI DENKI KABUSHIKI KAISHA) [JP/JP]; 〒100-8310 東京都千代田区丸の内二丁目2番3号 Tokyo (JP). 添付公開書類:  
— 国際調査報告書
- (72) 発明者; および
- (75) 発明者/出願人 (米国についてのみ): 松本 渉 (MATSUMOTO, Wataru) [JP/JP]; 〒100-8310 東京都千代田
- 2 文字コード及び他の略語については、定期発行される各 PCT ガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

(54) Title: LDPC CODE INSPECTION MATRIX GENERATION METHOD

(54) 発明の名称: LDPC 符号用検査行列生成方法



A...START  
S1...ASSUME RATE  
S2...HANDLE  $\lambda(x)$  AND  $\rho(x)$  AS VARIABLES AND SEARCH FOR  $\lambda(x)$  WHEN  $\sigma_n$  IS MAXIMUM  
B...END

(57) Abstract: An LDPC code inspection matrix generation method includes a step of analyzing the "Sum-Product algorithm" of the LDPC code by assuming that the logarithm likelihood ratio of the I/O data in the decoder can be approximated to the Gaussian distribution, a step of one linear programming to search for the most appropriate ensemble of line weight and row weight so that the Gaussian noise is maximum while the coding rate is fixed, and a step of generating an LDPC code inspection matrix according to the ensemble.

WO 03/056705 A1



---

(57) 要約:

本発明のLDPC符号用検査行列生成方法は、復号器における入出力データの対数尤度比がガウス分布に近似できると仮定してLDPC符号の「Sum-Productアルゴリズム」を解析し、符号化レートを固定した状態で、かつガウスノイズが最大になるように、行の重みと列の重みの最適なアンサンブルを1回の線形計画法で探索し、当該アンサンブルにしたがってLDPC符号用の検査行列を生成する。

## 明 細 書

## LDPC符号用検査行列生成方法

## 5 技術分野

本発明は、誤り訂正符号としてLDPC (Low-Density Parity-Check) 符号を採用した符号化器におけるLDPC符号用検査行列生成方法に関するものである。

## 背景技術

10 第13図は、LDPC符号化／復号システムを示す図である。第13図において、101は符号化器であり、102は変調器であり、103は通信路であり、104は復調器であり、105は復号器である。ここでは、従来のLDPC符号用検査行列生成方法を説明する前に、LDPC符号を使用した場合の符号化、復号の流れについて説明する。

15 まず、送信側の符号化器101では、後述する所定の方法で検査行列Hを生成する。そして、以下の条件に基づいて生成行列Gを求める。

$G : k \times n$  行列 ( $k$  : 情報長,  $n$  : 符号語長)

$GH^T = 0$  ( $T$ は転置行列)

その後、符号化器101では、情報長 $k$ のメッセージ( $m_1 m_2 \dots m_k$ )を受け取り、上記生成行列Gを用いて符号語Cを生成する。

$$\begin{aligned} C &= (m_1 m_2 \dots m_k) G \\ &= (c_1 c_2 \dots c_n) \quad (\text{ただし、} H(c_1 c_2 \dots c_n)^T = 0) \end{aligned}$$

そして、変調器102では、生成した符号語Cに対して、BPSK, QPSK, 多値QAMなどのデジタル変調を行い、送信する。

25 一方、受信側では、復調器104が、通信路103を介して受け取った変調信号に対して、BPSK, QPSK, 多値QAMなどのデジタル復調を行い、さらに、復号器105が、LDPC符号化された復調結果に対して、「sum-pr

product アルゴリズム」によるくり返し復号を実施し、推定結果（もとの  $m_1, m_2, \dots, m_k$  に対応）を出力する。

以下、従来の LDPC 符号用検査行列生成方法について説明する。LDPC 符号用の検査行列としては、たとえば、LDPC の発案者 Gallager により  
5 以下のような行列が提案されている（第 14 図参照）。

第 14 図に示す行列は、「1」と「0」の 2 値の行列で、「1」の部分塗りつぶしている。他の部分は全て「0」である。この行列は、1 行の「1」の数（これを行の重みと表現する）が 4 で、1 列の「1」の数（これを列の重みと表現する）が 3 であり、全ての列と行の重みが均一なため、これを一般に「Regular-LDPC 符号」と呼んでいる。また、Gallager の符号では、た  
10 えば、第 14 図に示すように、行列を 3 ブロックに分け、2 ブロック目と 3 ブロック目に対してランダム置換を行っている。

しかしながら、このランダム置換には、所定のルールが無いと、より特性の良好な符号を見つけるためには、計算機による時間のかかる探索を行わなければならない  
15 ならなかった。

そこで、たとえば、計算機探索によらなくても確定的に行列を生成でき、比較的安定した良好な特性を示す LDPC 符号として、ユークリッド幾何符号を用いる方法が、Y. Kou 等 (Y. Kou, S. Lin, and M. P. C. Fossorier, "Low Density Parity Check Codes Based on Finite Geometries: A Rediscovery," ISIT  
20 2000, pp. 200, Sorrento, Italy, June 25-30, 2000.) によって提案された。この方法では、規則的な ensemble (アンサンブル) で構成された「Regular-LDPC 符号」について説明されている。

ここでは、有限幾何符号の一種であるユークリッド幾何符号  $EG(2, 2^6)$  を用いて LDPC 符号の検査行列を生成する方法について提案され、誤り率  $10^{-4}$  点において、シャノン限界から 1.45 dB に接近した特性を得ている。第  
25 15 図は、たとえば、ユークリッド幾何符号  $EG(2, 2^2)$  の構成を示す図であり、行、列のそれぞれの重みが 4, 4 の「Regular-LDPC 符号」構

造をしている。

したがって、ユークリッド幾何符号EG (m, 2<sup>s</sup>) の場合、その特性は、以下のように規定される。

符号長:  $n = 2^{2s} - 1$

5 冗長ビット長:  $n - k = 3^s - 1$

情報長:  $k = 2^{2s} - 3^s$

最小距離:  $d_{\min} = 2^s + 1$

密度:  $r = 2^s / (2^{2s} - 1)$

第15図を見ても分かるように、ユークリッド幾何符号は、各行の「1」の配置が行毎に巡回シフトした構造になっており、符号が容易にかつ確定的に構成できる特長がある。

Y. Kouらによる検査行列の生成方法では、さらに、上記ユークリッド幾何符号に基づいて行と列の重みを変更し、行、列を必要に応じて拡張している。たとえば、EG (2, 2<sup>2</sup>) の列の重みを1/2に分離する場合、Y. Kouらの論文では、1列内に4つある重みを1つ置きに2個ずつ分離する。第16図は、列の重みを4から2に規則的に分離した例を示す図である。

一方、上記「Regular-LDPC符号」の特性よりも「Irregular-LDPC符号」の特性の方が良好であることが、Luby等 (M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Improved Low-Density Parity-Check Codes Using Irregular Graphs and Belief Propagation," Proceedings of 1998 IEEE International Symposium on Information Theory, p. 171, Cambridge, Mass., August 16-21, 1998.) により報告された。なお、上記「Irregular-LDPC符号」は、列と行の重みがそれぞれあるいはどちらか一方が均一でないLDPC符号を表す。

25 そして、それは、Richardson等 (T. J. Richardson and R. Urbanke, "The capacity of low-density parity-check codes under message-passing decoding," IEEE Trans. Inform. Theory, vol. 47, No. 2, pp. 599-618, Feb. 2001.

）、あるいはC h u n g 等 (S.-Y. Chung, T. J. Richardson, and R. Urbanke, "Analysis of Sum-Product Decoding of Low-Density Parity-Check Codes Using a Gaussian Approximation," IEEE Trans. Inform. Theory, vol. 47, No. 2, pp. 657-670, Feb. 2001.) によって理論的に解析された。

- 5      特に、C h u n g 等は、繰り返し復号器における入力と出力の対数尤度比 (L L R) がガウス分布に近似できると仮定してL D P C符号の「S u m - P r o d u c t アルゴリズム」を解析し、良好な行と列の重みのアンサンブルを求めている。

- しかしながら、たとえば、上記C h u n g 等による従来のL D P C符号用検査  
10    行列生成方法は、行内の「1」の点の数（後述するバリアブルノードの次数配分に相当）と、列内の「1」の点の数（後述するチェックノードの次数配分に相当）と、の両方を変数として、下記の（1）式（r a t e : 符号化率）が最大となるバリアブルノードの次数配分およびチェックノードの次数配分を求めている。  
すなわち、S N R (Signal to Noise Ratio) が最小となるアンサンブルを線形  
15    計画法により探索している。

$$\text{rate} = 1 - \frac{\int_0^1 \rho(x) dx}{\int_0^1 \lambda(x) dx} \quad \dots(1)$$

- そのため、上記「r a t e」の最大値により得られる検査行列が流動的になり、  
20    特性が安定しない、という問題があった。また、従来のL D P C符号用検査行列生成方法は、バリアブルノードの次数配分の導出とチェックノードの次数配分の導出とを所定回数にわたって繰り返し行っているため、探索処理にある程度の時間を要する、という問題もあった。

- 本発明は、上記に鑑みてなされたものであって、確定的でかつ特性が安定した  
25    L D P C符号用の検査行列を、短時間で容易に探索可能なL D P C符号用検査行列生成方法を提供することを目的としている。

## 発明の開示

本発明にかかるLDPC符号用検査行列生成方法にあつては、復号器における  
入出力データの対数尤度比がガウス分布に近似できると仮定してLDPC符号の  
「Sum-Productアルゴリズム」を解析することによって、誤りが0と  
5 なるSNRの限界(threshold)を求めることとし、さらに、符号化レートを固  
定した状態で、かつガウスノイズが最大になるように、行の重みと列の重みの最  
適なアンサンブル(thresholdが最小となるアンサンブル)を1回の線形計画法  
で探索し、当該アンサンブルにしたがってLDPC符号用の検査行列を生成する  
ことを特徴とする。

10 つぎの発明にかかるLDPC符号用検査行列生成方法にあつては、前記アンサ  
ンブルを探索後、当該探索結果に基づいてユークリッド幾何符号の各行または各  
列からランダムに「1」を抽出し、各行または各列を分割することによって、I  
rregular-LDPC符号の検査行列を生成することを特徴とする。

つぎの発明にかかるLDPC符号用検査行列生成方法にあつては、前記アンサ  
15 ンブルの重み配分を、重み単位の重み総数が整数で、かつ重み単位の重み総数の  
総和とユークリッド幾何符号の「1」の総数とが等しくなるように調整し、調整  
後のアンサンブルに基づいて分割処理を行うことを特徴とする。

つぎの発明にかかるLDPC符号用検査行列生成方法にあつては、基本のラン  
ダム系列のラテン方陣を作成し、ユークリッド幾何符号の第m列目をn分割する  
20 場合、上記ラテン方陣の第m行目のランダム系列をn分割し、当該n分割後の各  
ランダム系列を用いてユークリッド幾何符号の第m列目の「1」を抽出すること  
を特徴とする。

つぎの発明にかかるLDPC符号用検査行列生成方法にあつては、基本のラン  
ダム系列のラテン方陣を作成し、ユークリッド幾何符号の第m行目をn分割する  
25 場合、上記ラテン方陣の第m行目のランダム系列をn分割し、当該n分割後の各  
ランダム系列を用いてユークリッド幾何符号の第m行目の「1」を抽出すること  
を特徴とする。

つぎの発明にかかるLDPC符号用検査行列生成方法にあつては、基本のランダム系列のラテン方陣を複数個作成し、列方向に連結したラテン方陣群行列を用いて、ユークリッド幾何符号の第 $m$ 列目を $n$ 分割する場合、上記ラテン方陣群行列の第 $m$ 列目のランダム系列を $n$ 分割し、当該 $n$ 分割後の各ランダム系列を用いてユークリッド幾何符号の第 $m$ 列目の「1」を抽出することを特徴とする。

つぎの発明にかかるLDPC符号用検査行列生成方法にあつては、基本のランダム系列のラテン方陣を複数個作成し、列方向に連結したラテン方陣群行列を用いて、ユークリッド幾何符号の第 $m$ 行目を $n$ 分割する場合、上記ラテン方陣群行列の第 $m$ 列目のランダム系列を $n$ 分割し、当該 $n$ 分割後の各ランダム系列を用いてユークリッド幾何符号の第 $m$ 行目の「1」を抽出することを特徴とする。

#### 図面の簡単な説明

第1図は、実施の形態1のLDPC符号用検査行列生成方法を示すフローチャートであり、第2図は、 $rate = 0.5$ とした場合の生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ のアンサンブルを示す図であり、第3図は、 $EG(2, 2^5)$ を想定し、 $d_1 = 32$ とした場合の、分割テーブルを示す図であり、第4図は、重み配分調整用テーブルを示す図であり、第5図は、重み配分後の生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ のアンサンブルを示す図であり、第6図は、従来の分割手順を示す図であり、第7図は、分割前の $EG(2, 2^5)$ のグラフを示す図であり、第8図は、分割後の $EG(2, 2^5)$ のグラフを示す図であり、第9図は、 $E_b/N_0$ と誤り率特性との関係を示す図であり、第10図は、「Regular-LDPC符号」のアンサンブルを示す図であり、第11図は、「Irregular-LDPC符号」のアンサンブルを示す図であり、第12図は、ランダム系列のラテン方陣を示す図であり、第13図は、LDPC符号化/復号システムを示す図であり、第14図は、従来のLDPC符号用の検査行列を示す図であり、第15図は、ユークリッド幾何符号 $EG(2, 2^2)$ の構成を示す図であり、第16図は、列の重みを4から2に規則的に分離した例を示す図であり、



第17図は、ランダム系列のラテン方陣を示す図であり、第18図は、ランダム系列のラテン方陣を示す図である。

発明を実施するための最良の形態

- 5 以下に、本発明にかかるLDPC符号用検査行列生成方法の実施の形態を図面に基づいて詳細に説明する。なお、この実施の形態によりこの発明が限定されるものではない。

実施の形態1.

- 本実施の形態のLDPC符号用検査行列生成方法を説明する前に、本実施の形態のLDPC符号用検査行列生成方法を実現可能な符号化器の位置付け、および  
10 「Irregular-LDPC符号」用の従来の検査行列生成方法について説明する。なお、LDPC符号化／復号システムの構成については、先に説明した第13図と同様である。

- 送信側の符号化器101では、後述する本実施の形態のLDPC符号用検査行列生成方法で検査行列Hを生成する。そして、以下の条件に基づいて生成行列G  
15 を求める。

$G: k \times n$  行列 ( $k$ : 情報長,  $n$ : 符号語長)

$GH^T = 0$  ( $T$ は転置行列)

- その後、符号化器101では、情報長 $k$ のメッセージ( $m_1 m_2 \dots m_k$ )を受け取り、上記生成行列Gを用いて符号語Cを生成する。  
20

$$C = (m_1 m_2 \dots m_k) G$$

$$= (c_1 c_2 \dots c_n) \quad (\text{ただし、} H(c_1 c_2 \dots c_n)^T = 0)$$

そして、変調器102では、生成した符号語Cに対して、BPSK, QPSK, 多値QAMなどのデジタル変調を行い、送信する。

- 25 一方、受信側では、復調器104が、通信路103を介して受け取った変調信号に対して、BPSK, QPSK, 多値QAMなどのデジタル復調を行い、さらに、復号器105が、LDPC符号化された復調結果に対して、「sum-pr

product アルゴリズム」による繰り返し復号を実施し、推定結果（もとの  $m_1, m_2, \dots, m_k$  に対応）を出力する。

つぎに、Chung 等 (S.-Y. Chung, T. J. Richardson, and R. Urbanke, "Analysis of Sum-Product Decoding of Low-Density Parity-Check Codes Using a Gaussian Approximation," IEEE Trans. Inform. Theory, vol. 47, No. 2, pp. 657-670, Feb. 2001.) によって理論的に解析された、「Irregular-LDPC 符号」用の従来の検査行列生成方法について詳細に説明する。ここでは、繰り返し復号器における入力と出力の対数尤度比 (LLR) がガウス分布に近似できると仮定して LDPC 符号の「Sum-Product アルゴリズム」を解析し、良好な行と列の重みのアンサンブルを求めている。

なお、上記論文に記述された LDPC 符号用検査行列生成方法であるガウス近似法 (Gaussian Approximation) では、前提として、検査行列における行内の「1」の点をバリエブルノードと定義し、列内の「1」の点をチェックノードと定義する。

まず、チェックノードからバリエブルノードへの LLR メッセージ伝搬を解析する。  $0 < s < \infty$  と  $0 \leq t < \infty$  という条件において、以下の関数 (2) 式を定義する。なお、  $s = m_{u0}$  は  $u_0$  の平均値であり、  $u_0$  は分散値  $\sigma_u^2$  のガウスノイズを含む伝送路を経由して受信した信号の対数尤度比 (LLR) であり、  $t$  は所定の繰り返しの時点におけるチェックノードの LLR 出力値のアンサンブル平均である。

$$f_j(s, t) = \phi^{-1} \left( 1 - \left[ 1 - \sum_{i=2}^{d_j} \lambda_i \phi(s + (i-1)t) \right]^{j-1} \right)$$

$$f(s, t) = \sum_{j=2}^{d_1} \rho_j f_j(s, t) \quad \dots (2)$$

なお、上記  $\lambda(x)$  および  $\rho(x)$  は、それぞれバリエブルノードおよびチェックノードの次数配分（バリエブルノードとチェックノードの各 1 行、各 1 列内の「1」の数を次数と表現する）の生成関数を表し、(3) 式および (4) 式

のように表すことができる。また、 $\lambda_i$ と $\rho_i$ は、それぞれ次数 $i$ のバリエブルノードとチェックノードに属するエッジの比率を表す。また、 $d_1$ は最大バリエブルノードの次数であり、 $d_r$ は最大チェックノードの次数である。

$$\lambda(x) = \sum_{i=2}^{d_1} \lambda_i x^{i-1} \quad \dots(3)$$

5

$$\rho(x) = \sum_{i=2}^{d_r} \rho_i x^{i-1} \quad \dots(4)$$

ただし、 $\phi(x)$ は下記(5)式のように定義する。

$$\phi(x) = \begin{cases} 1 - \frac{1}{\sqrt{4\pi x}} \int_0^x \tanh \frac{u}{2} \cdot e^{-\frac{(u-x)^2}{4x}} du & \text{if } x > 0 \\ 1 & \text{if } x \leq 0 \end{cases} \quad \dots(5)$$

10

そして、(2)式は、等価的に下記(6)式と表すことができる。

$$t_1 = f(s, t_{1-1}) \quad \dots(6)$$

15

なお、 $t_1$ は1番目の繰り返し時点におけるチェックノードのLLR出力値のアンサンブル平均である。

ここで、誤りが0となりうるSNRの限界(threshold)を求めるための条件は、 $1 \rightarrow \infty$ のときに $t_1(s) \rightarrow \infty$ ( $R^+$ と表現する)となることであり、この条件を満たすためには、以下の条件(7)式を満たす必要がある。

20

$$t < f(s, t), \text{ 全ての } t \in R^+ \quad \dots(7)$$

つぎに、バリエブルノードからチェックノードへのLLRメッセージ伝搬を解析する。 $0 < s < \infty$ と $0 < r \leq 1$ という条件において、以下の関数(8)式を定義する。なお、 $r$ の初期値 $r_0$ は $\phi(s)$ である。

25

$$h_i(s, r) = \phi \left( s + (i-1) \sum_{j=2}^{d_i} \rho_j \phi(1 - (1-r)^{j-1}) \right)$$

$$h(s, r) = \sum_{i=2}^{d_1} \lambda_i h_i(s, r) \quad \dots(8)$$

5       そして、(8) 式は、等価的に下記 (9) 式と表すことができる。

$$r_i = h(s, r_{i-1}) \quad \dots(9)$$

ここで、誤りが 0 となりうる SNR の限界 (threshold) を求めるための条件  
 は、 $r_1(s) \rightarrow 0$  となることであり、この条件を満たすためには、以下の条件  
 10       (10) 式を満たす必要がある。

$$r > h(s, r), \text{ 全ての } r \in (0, \phi(s)) \quad \dots(10)$$

さらに、上記 Chung 等の論文では、上記式を用いて以下の手順でバリエーション  
 ノードとチェックノードの最適な次数を探索している。

15       (1) 生成関数  $\lambda(x)$  とガウスノイズ  $\sigma_n$  が与えられていると仮定し、生成関  
 数  $\rho(x)$  を変数として、前述した (1) 式が最大となる点を探索する。なお、  
 この探索における拘束条件は、 $\rho(1) = 1$  と正規化することと、上記 (7) 式  
 を満たすことである。

20       (2) 生成関数  $\rho(x)$  とガウスノイズ  $\sigma_n$  が与えられていると仮定し (たとえ  
 ば、(1) の結果より得られる値)、生成関数  $\lambda(x)$  を変数として、(1)  
 式が最大となる点を探索する。なお、この探索における拘束条件は、 $\lambda(1) =$   
 $1$  と正規化することと、上記 (10) 式を満たすことである。

25       (3) 最大「rate」を求めるために、上記 (1) と上記 (2) を繰り返し実  
 行し、生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のより良好なアンサンブルを線形  
 計画法で探索する。

(4) 最後に、ガウスノイズ  $\sigma_n$  より信号電力を 1 と正規化して、SNR の限界  
 (threshold) を求める。

$$\text{threshold(dB)} = -10 * \log_{10}(2 * \sigma_n^2) \quad \dots(11)$$

しかしながら、上記 Chung 等の論文では、「rate (符号化率)」の最大値により得られる検査行列が流動的になり、設計において仕様として固定される rate が変動してしまうため、実設計にむかない、という問題があった。また、上記ガウス近似法では、バリアブルノードの次数配分の導出とチェックノードの次数配分の導出とを所定回数にわたって繰り返し行っているため、探索処理にある程度の時間を要する、という問題もあった。

そこで、本実施の形態においては、確定的でかつ特性が安定した「Irregular-LDPC符号」用の検査行列を、短時間で容易に探索する。第1図は、実施の形態1のLDPC符号用検査行列生成方法を示すフローチャートである。

(1) 「rate」が与えられているものと仮定する。すなわち、要求「rate」を固定する。実際の設計では、目標「rate」が予め指定されている場合が多いためである。

(2) 生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  を同時に変数として扱い、ガウスノイズ  $\sigma_n$  が最大になるように、線形計画法で最適な生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  を探索する。この探索における拘束条件は、 $\lambda(1) = 1$ ,  $\rho(1) = 1$  と正規化し、さらに上記(10)式を満たすことである。

このように、本実施の形態では、上記(9)式と上記(10)式を満たす生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  を1回の線形計画法で求めることとしたため、上記論文のように、生成関数  $\lambda(x)$  の導出と生成関数  $\rho(x)$  の導出を繰り返し実行し、双方の最適値を求める方法よりも、容易かつ短時間に、確定的でかつ特性が安定したLDPC符号用の検査行列を生成することができる。

実施の形態2.

実施の形態2では、前述の実施の形態1においてユークリッド幾何符号を採用し、1行あるいは1列の「1」の配置を分割することによって、「Irregular-LDPC符号」の検査行列を生成する。

まず、実施の形態1におけるLDPC符号用検査行列生成方法により、生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンブルを導出する。第2図は、 $rate = 0.5$  とした場合の生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンブルを示す図である。なお、 $\sigma_{GA}$  はガウス近似法により導出した「threshold」時の  
 5 ノイズ分散値を表し、 $SNR_{norm}(GA)$  はガウス近似法により導出した「threshold」のSNRとシャノン限界のSNRとの差分を表し、 $x$  は重みを表し、 $\lambda_x$  および  $\rho_x$  はそれぞれパリアブルノードとチェックノードの重み配分を表す。

また、基準になるユークリッド幾何符号は、 $EG(2, 2^5)$  を想定し、 $d_1 = 32$  とした。また、重み配分  $\lambda_x$  の  $x$  の値および重み配分  $\rho_x$  の  $x$  の値は、組み  
 10 合わせで  $32(d_1)$  を構成できる値とする。

第3図は、 $EG(2, 2^5)$  を想定し、 $d_1 = 32$  とした場合の、分割テーブルを示す図である。第3図に示すとおり、第2図の  $x$  は、組み合わせにより必ず  $32$  となる。たとえば、図示の  $7 \times 4$  と  $2 \times 2$  の組み合わせは、重みが  $32$  の1列を、重みが7の4列と重みが2の2列に分割可能なことを表している。このよう  
 15 に、 $EG(2, 2^5)$  の符号を基本として、第3図のように重みが  $32$  の各行列を適切に分割すると、「Irregular-LDPC符号」の検査行列を構成することができる。

ここで、分割処理を行う前に、第2図に示す生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンブルの重み配分を以下の手順で調整する。第4図は、重み配分  
 20 調整用テーブルを示す図である。なお、 $EG(2, 2^5)$  のユークリッド幾何符号は、 $1023$  行  $\times$   $1023$  列で構成される。

(1) ガウス近似法で求めた生成関数  $\lambda(x)$  と生成関数  $\rho(x)$  のアンサンブル(表1参照)をテーブルの2列目と3列目に設定する。

(2) 重み配分  $\lambda_x$  および  $\rho_x$  (3列目)と、 $EG(2, 2^5)$  における全行列の  
 25 「1」の総数  $TP = 32736$  と、を乗算し、重み単位の重み総数を求め、さらに、当該重み単位の重み総数とその総和を4列目に設定する。

(3) 重み単位の重み総数(4列目)を対応する重み  $x$  で割り、重み単位の総列

数を求め、それを5列目に設定する。

(4) 重み単位の総列数が小数点以下を含む場合、丸め処理(四捨五入, 切上げ, 切捨て等)を行い、その結果を6列目に設定する。

(5) 丸め処理後の重み単位の総列数(6列目)と対応する重み $x$ とを乗算し、  
5 丸め処理後の重み単位の重み総数を求め、それを7列目に設定する。そして、各重み総数の総和(7列目の合計の行)が行列内の「1」の総数( $TP = 3\ 2\ 7\ 3\ 6$ )と等しいかどうかを確認する。

(6) 行列内の「1」の総数に等しくない場合、丸め処理後の重み単位の重み総数(7列目)を整数単位で調整し、その結果を8列目に設定する。この場合、8  
10 列目の総和が、行列内の「1」の総数( $TP = 3\ 2\ 7\ 3\ 6$ )に等しくなるように調整する。

(7) 調整後の重み単位の重み総数(8列目)を対応する重み $x$ で割り、調整後の重み単位の総列数を求め、それを9列目に設定する。調整後の各重みの配分(11列目)は、可能な限りガウス近似法で求めた値(3列目)に近い値にする。

15 第5図は、重み配分後の生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ のアンサンブルを示す図である。

つぎに、ユークリッド幾何符号における1行あるいは1列の分割手順について説明する。

たとえば、分割手順に関して、Y. Kou等の論文では、規則的に分割する方法を提示している。第6図は、上記論文における分割手順を示す図である。まず、  
20 第6図に示すように行列のナンバリングを行う。ここでは、列番号を左端から順に1, 2, 3, ...とし、行番号を上から順に1, 2, 3, ...とする。そして、たとえば、32点×1列を8点×4列に分割する場合、下記(12)式にしたがって規則的に分割する。

25

$$S_m(n) = B_1(m + 4 \cdot n) \quad \dots(12)$$

なお、 $m = 1, 2, 3, 4$ とし、 $n = 0, 1, 2, 3, 4, 5, 6, 7$ とし、

1 はEG (2, 2<sup>5</sup>) の列番号を表す。また、B<sub>1</sub> (x) はEG (2, 2<sup>5</sup>) の1列目の「1」の位置を表し、S<sub>m</sub> (n) は分割後の行列のm列目の「1」の位置を表す。

5 具体的にいうと、EG (2, 2<sup>5</sup>) における1列中の「1」の位置を示す行番号は、

$$B_1(x) = \{1 \ 32 \ 114 \ 136 \ 149 \ 223 \ 260 \ 382 \ 402 \ 438 \ 467 \ 507 \ 574 \ 579 \ 588 \ 622 \ 634 \ 637 \ 638 \ 676 \ 717 \ 728 \ 790 \ 851 \ 861 \ 879 \ 947 \ 954 \ 971 \ 977 \ 979 \ 998\}$$

となり、その結果、分割後の行列における1～4列目の「1」の位置を示す行番号は、B<sub>1</sub> (x) から「1」の番号が規則的に抽出され、

10  $S_1(n) = \{1 \ 149 \ 402 \ 574 \ 634 \ 717 \ 861 \ 971\}$

$$S_2(n) = \{32 \ 223 \ 438 \ 579 \ 637 \ 728 \ 879 \ 977\}$$

$$S_3(n) = \{114 \ 260 \ 467 \ 588 \ 638 \ 790 \ 947 \ 979\}$$

$$S_4(n) = \{136 \ 382 \ 507 \ 622 \ 676 \ 851 \ 954 \ 998\}$$

となる。すなわち、3 2点×1列が8点×4列に分割される。

15 一方、本実施の形態におけるユークリッド幾何符号の分割処理は、上記のように規則的に分割するのではなく、B<sub>1</sub> (x) から「1」の番号をランダムに抽出する。なお、この抽出処理は、ランダム性が保持されるのであればどのような方法を用いてもよい。

したがって、分割後の行列のm列目の「1」の位置の一例をR<sub>m</sub> (n) とした場合、R<sub>m</sub> (n) は、

20  $R_1(n) = \{1 \ 114 \ 574 \ 637 \ 851 \ 879 \ 977 \ 979\}$

$$R_2(n) = \{32 \ 136 \ 402 \ 467 \ 588 \ 728 \ 861 \ 971\}$$

$$R_3(n) = \{149 \ 260 \ 382 \ 438 \ 579 \ 638 \ 717 \ 998\}$$

$$R_4(n) = \{223 \ 507 \ 622 \ 634 \ 676 \ 790 \ 947 \ 954\}$$

25 となる。

上記のような本実施の形態の分割手順をグラフ上で表現すると、以下のように表現することができる。第7図は、分割前のEG (2, 2<sup>5</sup>) のグラフを示す図



である。なお、両ノードを結ぶ線はエッジと表現する。第7図では、分割前の1023行×1023列（各行列の重みがそれぞれ32）のユークリット幾何符号を表現している。また、第8図は、 $EG(2, 2^5)$ のエッジをランダムに選択した、分割後のグラフを示す図である。

- 5       ここで、上記で説明したLDPC符号の特性を比較する。第9図は、 $E_b/N_0$ （情報1ビットあたりの信号電力対ノイズ電力比）と誤り率特性（BER）との関係を示す図である。なお、繰り返し回数は50回で、復号法は「Sum-Productアルゴリズム」である。

        なお、図中“Simple regular extended  $EG(2, 2^5)$ ”は、Y. Kou等の発案による $EG(2, 2^5)$ の規則的な列の分割を実施した場合の、 $rate=0.5$ の「Regular-LDPC符号」であり、“Random regular extended  $EG(2, 2^5)$ ”は、本実施の形態による $EG(2, 2^5)$ のランダムな列の分割を実施した場合の、 $rate=0.5$ の「Regular-LDPC符号」である。第10図は、これらの「Regular-LDPC符号」のアンサンブルを示す図である。

- 15       また、図中“Simple irregular extended  $EG(2, 2^5)$ ”は、実施の形態1の方法によって特定されたアンサンブルに対して、Y. Kou等の発案による $EG(2, 2^5)$ の規則的な列の分割を実施した場合の、 $rate=0.5$ の「Irregular-LDPC符号」であり、“Random irregular extended  $EG(2, 2^5)$ ”は、実施の形態1の方法によって特定されたアンサンブルに対して、本実施の形態による $EG(2, 2^5)$ のランダムな列の分割を実施した場合の、 $rate=0.5$ の「Irregular-LDPC符号」である。第11図は、これらの「Irregular-LDPC符号」のアンサンブルを示す図である。

        第9図からわかるように、同一レートでは、「Regular-LDPC符号」より「Irregular-LDPC符号」のほうが性能がよい。また、Y. Kou等の論文のような規則的な分割では、「Irregular-LDPC符号」であっても大幅な改善は見込めないが、本実施の形態のランダムな分割を実施すると性能が画期的に改善される。

つぎに、上記ランダム分割の一例を詳細に説明する。ここでは、ランダム分割を行う場合のランダム系列を容易かつ確定的に生成する。この方法による利点は、送信側と受信側が同じランダム系列を生成できることにある。これは、現実のシステムではきわめて重要となる。また、符号特性の条件が正確に規定できる、という利点もある。

(1) 基本のランダム系列を作成する。

以下に、ランダム系列作成の一例を記述する。ここでは、説明の便宜上、ユークリッド幾何符号EG (2, 2<sup>4</sup>) を用いる。ユークリッド幾何符号EG (2, 2<sup>4</sup>) の場合、1行に存在する「1」の数は2<sup>4</sup>個である。

10 Pを $P \geq 2^s$ を満たす最小の素数とした場合、たとえば、2<sup>4</sup>のときは $P = 17$ となる。ここで、系列長 $P - 1 = 16$ の基本のランダム系列C (i) を(13)式にしたがって作成する。

$$C(1) = 1$$

$$C(i+1) = G_0 \times C(i) \mod P \quad \dots (13)$$

15 ただし、 $i = 1, \dots, P - 1$ とし、 $G_0$ はガロア体GF (P) の原始元である。その結果、C (i) は、

$$C(i) = \{1 \ 3 \ 9 \ 10 \ 13 \ 5 \ 15 \ 11 \ 16 \ 14 \ 8 \ 7 \ 4 \ 12 \ 2 \ 6\}$$

となる。なお、2<sup>4</sup>の場合は、Pが2<sup>4</sup>+1となり、ランダム系列長が16になるので問題ないが、2<sup>5</sup>の場合等は、Pが2<sup>5</sup>+1以上になりランダム系列長が2<sup>5</sup>を超えてしまう。そのような場合は、2<sup>5</sup>を超える数値をランダム系列から削除することで対応する。

そして、ランダム系列C (i) を巡回シフトして、ランダム系列のラテン方陣を生成する。第12図は、ランダム系列のラテン方陣を示す図である。

(2) ランダム系列のラテン方陣を用いて、ユークリッド幾何符号EG (2, 2<sup>4</sup>) の1列目の「1」の位置を示す行番号の配列B<sub>1</sub> (x) から、「1」の番号をランダムに抽出し、ユークリッド幾何符号の第1列目を分割する。なお、上記配列B<sub>1</sub> (x) は、

$$B_1(x) = \{1 \ 14 \ 16 \ 19 \ 45 \ 49 \ 55 \ 107 \ 115 \ 126 \ 127 \ 182 \ 210 \ 224 \ 231 \ 247\}$$

である。

たとえば、ユークリッド幾何符号の第1列目を4分割する場合（各列の重みが4となる）、上記ランダム系列のラテン方陣の1行目から、4つのランダム系列

5  $L_1(n) \sim L_4(n)$  を抽出する。

$$L_1(n) = \{1, 3, 9, 10\}$$

$$L_2(n) = \{13, 5, 15, 11\}$$

$$L_3(n) = \{16, 14, 8, 7\}$$

$$L_4(n) = \{4, 12, 2, 6\}$$

10 そして、4つのランダム系列  $L_1(n) \sim L_4(n)$  を用いて、配列  $B_1(x)$  から、「1」の番号をランダムに抽出する。その結果、 $R(L_1(n)) \sim R(L_4(n))$  は、

$$R(L_1(n)) = \{1, 16, 115, 126\}$$

$$R(L_2(n)) = \{210, 45, 231, 127\}$$

15  $R(L_3(n)) = \{247, 224, 107, 55\}$

$$R(L_4(n)) = \{19, 182, 14, 49\}$$

となる。

（3）同様に、ランダム系列のラテン方陣を用いて、ユークリッド幾何符号 EG

（2, 2<sup>4</sup>）の2列目の「1」の位置を示す行番号の配列  $B_2(x)$  から、「1」

20 の番号をランダムに抽出し、ユークリッド幾何符号の第2列目を分割する。なお、上記配列  $B_2(x)$  は、

$$B_2(x) = \{2 \ 15 \ 17 \ 20 \ 46 \ 50 \ 56 \ 108 \ 116 \ 127 \ 128 \ 183 \ 211 \ 225 \ 232 \ 248\}$$

である。

たとえば、ユークリッド幾何符号の第2列目を4分割する場合（各列の重みが4となる）、上記ランダム系列のラテン方陣の2行目から、4つのランダム系列

25  $L_1(n) \sim L_4(n)$  を抽出する。

$$L_5(n) = \{3, 9, 10, 13\}$$

$$L_6(n) = \{5, 15, 11, 16\}$$

$$L_7(n) = \{14, 8, 7, 4\}$$

$$L_8(n) = \{12, 2, 6, 1\}$$

そして、4つのランダム系列 $L_5(n) \sim L_8(n)$ を用いて、配列 $B_2(x)$ から、「1」の番号をランダムに抽出する。その結果、 $R(L_5(n)) \sim R(L_8(n))$ は、

$$R(L_5(n)) = \{17, 116, 127, 211\}$$

$$R(L_6(n)) = \{46, 232, 128, 248\}$$

$$R(L_7(n)) = \{225, 108, 56, 20\}$$

$$R(L_8(n)) = \{183, 15, 50, 2\}$$

となる。以降、同様の手順でユークリッド幾何符号の全ての列を分割する。

(4) なお、上記ランダム系列のラテン方陣でランダム系列が不足するような場合は、(13)式の基本のランダム系列の $G_0$ に原始元より大きくかつ $2^s$ 以下の素数を代入して再度基本のランダム系列を作り、同様の手順で分割する。

つぎに、もう一つのランダム系列のラテン方陣の作成法について説明する。ここでは、説明の便宜上、ユークリッド幾何符号 $EG(2, 2^5)$ を用いる。ユークリッド幾何符号 $EG(2, 2^5)$ の場合、1行に存在する「1」の数は $2^5$ 個である。

上記例では、 $EG(2, 2^4)$ のため16(行数)×16(列数)のラテン方陣を作成したが、これでは不足する場合がある。ここでは、 $EG(2, 2^5)$ を例にとり、32(行数)×960(列数)まで拡張する。行列のサイズは、32(行数)×32(列数)のラテン方陣の組を30組列方向に並べたことにより決められる。

また、上記の例では、 $P$ を $P \geq 2^s$ を満たす最小の素数としたが、ここでは、 $P$ を $P \geq 2^s$ を満たす最大の素数とする。たとえば、 $2^5$ のときは $P=31$ となる。

ここで、系列長 $P=31$ の基本のランダム系列 $C(i)$ を(14)式に従って作成する。

$$C(1) = 0$$

$$C(i+1) = G_0 \times C(i) \mod P \quad \dots (14)$$

ただし、 $i = 1, \dots, P-1$  とし、 $G_0$  はガロア体  $GF(P)$  の原始元である。

その結果、 $C(i)$  は、

$$C(i) = \{0, 1, 3, 9, 27, 19, 26, 17, 20, 29, 25, 13, 8, 24, 10, 30, 28, 22, 4, 12, 5, 15, 14, 11, 2, 6, 18, 23, 7, 21\}$$

となる。

つぎに以下の操作をする。

$$C(i) = C(i) + 1, \quad C(32) = 32$$

その結果、 $C(i)$  は、

$$C(i) = \{1, 2, 4, 10, 28, 20, 27, 17, 18, 21, 30, 26, 14, 9, 25, 11, 31, 29, 23, 5, 13, 6, 16, 15, 12, 3, 7, 19, 24, 8, 22, 32\}$$

となる。これを第17図の左側の太枠内に示し、基本のランダム系列とする。

つぎに、 $C(i)$  のランダム系列をある間隔  $S(j)$  ,  $j = 1, 2, \dots, P-1$

1 で読む方法について説明する。この間隔は、 $P-1$  個生成可能である。ここでは  $P-1 = 30$  である。

$$S(j) = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 2, 3, 24, 25, 26, 27, 28, 29, 30\}$$

ランダム系列を一定の間隔で飛ばし読みする系列を  $LB_j(i)$  とすると、

$$LB_j(i) = ((S(j) * i) \mod P) + 1$$

ただし、 $j = 1, 2, \dots, P-1$  であり、 $i = 1, \dots, P-1$  である。

たとえば  $j = 1$  の場合、

$$LB_1(i) = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 2, 3, 24, 25, 26, 27, 28, 29, 30, 31\}$$

となる。

ここで、ランダム系列の系列数 32 に対し、1 と  $P$  から  $2^*$  までの整数が不足している。この場合は 1, 32 が不足している。

$$LB_j(j) = 32, LB_j(32-j) = 1$$

を挿入する。

この結果、

$$LB_1(i) = \{32, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 2, 23, 24, 25, 26, 27, 28, 29, 30, 1, 31\}$$

となる。

同様に  $j = 2$  の場合、

$$LB_2(i) = \{3, 32, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 1, 28, 30\}$$

ここからラテン方陣を作成する。 $L_{jq}(i)$  は  $j$  番目のラテン方陣の  $q$  列目の系列である。 $L_{jq}(i) = LB_j(i)$  とすると、たとえば、

$$L_{11}(i) = C(LB_1(i)) = \{32, 2, 4, 10, 28, 20, 27, 17, 18, 21, 30, 26, 14, 9, 25, 11, 31, 29, 23, 5, 13, 6, 16, 15, 12, 3, 7, 19, 24, 8, 1, 22\}$$

となる。これは、第18図の1列目が  $L_{11}(i)$  に相当する。

このランダム系列  $L_{11}(i)$  を巡回シフトして  $L_{1q}(i)$  の  $32$  (行数)  $\times 32$  (列数) のラテン方陣を作成する。同様の手順で、ラテン方陣を  $L_{2q}(i)$ ,  $L_{3q}(i)$ , ...,  $L_{30q}(i)$  まで作成し、 $32$  (行数)  $\times (32 \times 30)$  (列数) のラテン方陣の組み合わせを作成する。

ここで、具体例を説明する。ユークリッド幾何符号  $EG(2, 2^5)$  の1列目の「1」の位置を示す行番号の配列  $B_1(x)$  から「1」の番号をランダムに抽出し、ユークリッド幾何符号の第1列目を分解する。なお、上記配列  $B_1(x)$  は、

$$B_1(x) = \{1\ 32\ 114\ 136\ 149\ 223\ 260\ 382\ 402\ 438\ 467\ 507\ 574\ 579\ 588\ 622\ 634\ 637\ 638\ 676\ 717\ 728\ 790\ 851\ 861\ 879\ 947\ 954\ 971\ 977\ 979\ 998\}$$

である。

たとえば、ユークリッド幾何符号の第1列目を4分割する場合(各列の重みが8となる)、上記ランダム系列のラテン方陣組み合わせ  $L_{jq}(i)$  の1番目の

ラテン方阵の1列目から4つのランダム系列 $L_1(n) \sim L_4(n)$ を抽出する。

$$L_1(n) = \{32, 2, 4, 10, 28, 20, 27, 17\}$$

$$L_2(n) = \{18, 21, 30, 26, 14, 9, 25, 11\}$$

$$L_3(n) = \{31, 29, 23, 5, 13, 6, 16, 15\}$$

5  $L_4(n) = \{12, 3, 7, 19, 24, 8, 1, 22\}$

そして、4つのランダム系列 $L_1(n) \sim L_4(n)$ を用いて、配列 $B_1(x)$ から「1」の番号をランダムに抽出する。その結果 $R(L_1(n)) \sim R(L_4(n))$ は、

$$R(L_1(n)) = \{998, 32, 136, 438, 954, 676, 947, 634\}$$

10  $R(L_2(n)) = \{637, 717, 977, 879, 579, 402, 861, 467\}$

$$R(L_3(n)) = \{979, 971, 790, 149, 574, 223, 622, 588\}$$

$$R(L_4(n)) = \{507, 114, 260, 638, 851, 382, 1, 728\}$$

となる。以降、同様の手順でユークリッド幾何符号の全ての列を分割する。

以上、説明したとおり、本発明によれば、生成関数 $\lambda(x)$ と生成関数 $\rho(x)$ を1回の線形計画法で求めることとしたため、上記論文のように、生成関数 $\lambda(x)$ の導出と生成関数 $\rho(x)$ の導出を繰り返し実行し、双方の最適値を求める方法よりも、容易かつ短時間に、確定的でかつ特性が安定したLDPC符号用の検査行列を生成することができる、という効果を奏する。

つぎの発明によれば、同一レートでは「Regular-LDPC符号」よりも良好な特性を得ることができる、という効果を奏する。また、規則的な分割では、「Irregular-LDPC符号」であっても大幅に特性を改善することはできないが、ランダムな分割を実施すると、特性を画期的に改善することができる、という効果を奏する。

つぎの発明によれば、重み配分を、重み単位の重み総数が整数で、かつ重み単位の重み総数の総和とユークリッド幾何符号の「1」の総数とが等しくなるように調整することによって、より高精度な分割処理を実現できる、という効果を奏する。

つぎの発明によれば、送信側と受信側が同じランダム系列を生成することができる、という効果を奏する。また、ランダム系列のラテン方陣を作成することによって、符号特性の条件を正確に規定できる、という効果を奏する。

- 5 つぎの発明によれば、送信側と受信側が同じランダム系列を生成することができる、という効果を奏する。また、ランダム系列のラテン方陣を作成することによって、符号特性の条件を正確に規定できる、という効果を奏する。

つぎの発明によれば、送信側と受信側が同じランダム系列を生成することができる、という効果を奏する。また、ランダム系列のラテン方陣を複数個作成することによって、符号特性の条件を正確に規定できる、という効果を奏する。

- 10 つぎの発明によれば、送信側と受信側が同じランダム系列を生成することができる、という効果を奏する。また、ランダム系列のラテン方陣を複数個作成することによって、符号特性の条件を正確に規定できる、という効果を奏する。

#### 産業上の利用可能性

- 15 以上のように、本発明にかかるLDPC符号用検査行列生成方法は、誤り制御技術としてLDPC符号を採用するLDPC符号化／復号システムに有用であり、特に、確定的で特性が安定した「Irregular-LDPC符号」を生成するための符号化器および当該符号化器を備えた通信装置に適している。



## 請 求 の 範 囲

1. 復号器における入出力データの対数尤度比がガウス分布に近似できると仮定してLDPC (Low-Density Parity-Check) 符号の「Sum-Product  
5 アルゴリズム」を解析することによって、誤りが0となるSNRの限界 (threshold) を求めるLDPC符号用検査行列生成方法において、  
符号化レートを固定した状態で、かつガウスノイズが最大になるように、行の重みと列の重みの最適なアンサンブル (threshold が最小となるアンサンブル) を1回の線形計画法で探索し、当該アンサンブルにしたがってLDPC符号用の  
10 検査行列を生成することを特徴とするLDPC符号用検査行列生成方法。  
  
2. 前記アンサンブルを探索後、当該探索結果に基づいてユークリッド幾何符号の各行または各列からランダムに「1」を抽出し、各行または各列を分割することによって、Irregular-LDPC符号の検査行列を生成することを  
15 特徴とする請求の範囲第1項に記載のLDPC符号用検査行列生成方法。  
  
3. 前記アンサンブルの重み配分を、重み単位の重み総数が整数で、かつ重み単位の重み総数の総和とユークリッド幾何符号の「1」の総数とが等しくなるように調整し、調整後のアンサンブルに基づいて分割処理を行うことを特徴とする  
20 請求の範囲第2項に記載のLDPC符号用検査行列生成方法。  
  
4. 基本のランダム系列のラテン方陣を作成し、  
ユークリッド幾何符号の第m列目をn分割する場合、上記ラテン方陣の第m行目のランダム系列をn分割し、当該n分割後の各ランダム系列を用いてユークリ  
25 ュッド幾何符号の第m列目の「1」を抽出することを特徴とする請求の範囲第3項に記載のLDPC符号用検査行列生成方法。

5. 基本のランダム系列のラテン方陣を作成し、

ユークリッド幾何符号の第 $m$ 行目を $n$ 分割する場合、上記ラテン方陣の第 $m$ 行目のランダム系列を $n$ 分割し、当該 $n$ 分割後の各ランダム系列を用いてユークリッド幾何符号の第 $m$ 行目の「1」を抽出することを特徴とする請求の範囲第3項

5 に記載のLDPC符号用検査行列生成方法。

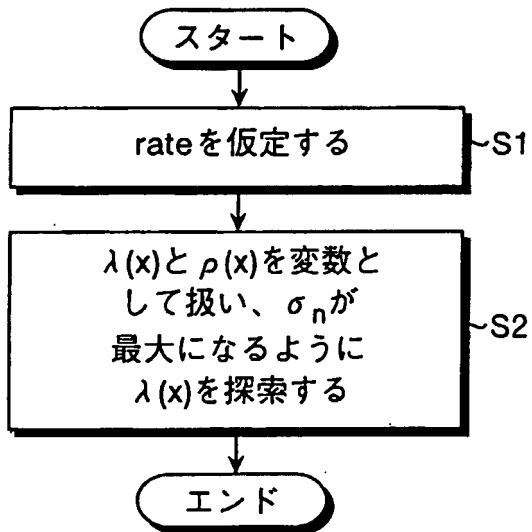
6. 基本のランダム系列のラテン方陣を複数個作成し、

列方向に連結したラテン方陣群行列を用いて、ユークリッド幾何符号の第 $m$ 列目を $n$ 分割する場合、上記ラテン方陣群行列の第 $m$ 列目のランダム系列を $n$ 分割し、当該 $n$ 分割後の各ランダム系列を用いてユークリッド幾何符号の第 $m$ 列目の「1」を抽出することを特徴とする請求の範囲第3項に記載のLDPC符号用検査行列生成方法。

7. 基本のランダム系列のラテン方陣を複数個作成し、

15 列方向に連結したラテン方陣群行列を用いて、ユークリッド幾何符号の第 $m$ 行目を $n$ 分割する場合、上記ラテン方陣群行列の第 $m$ 列目のランダム系列を $n$ 分割し、当該 $n$ 分割後の各ランダム系列を用いてユークリッド幾何符号の第 $m$ 行目の「1」を抽出することを特徴とする請求の範囲第3項に記載のLDPC符号用検査行列生成方法。

## 第1図



## 第2図

$d_t$	32	
rate	0.5	
	$x$	$\lambda_x$
	2	0.178783
	3	0.149148
	6	0.019896
	7	0.220144
	8	0.005388
	32	0.426641
	$x$	$\rho_x$
	10	0.333333
	11	0.666667
$\sigma_{GA}$	0.962089	
$SNR_{norm}(GA)$	0.1486 dB	

## 第 3 図

2x16	3x10	6x5	7x4	8x4	32x1
	2x1	2x1	2x2		

第4図

1	2	3	4	5	6	7	8	9	10	11
	x	$\lambda x$							x	$\lambda x$
チェック	2	0.178783	5852.64	2926.32	2926	5852	5846	2923	2	0.178580156
ノード	3	0.149148	4882.509	1627.503	1628	4884	4890	1630	3	0.149376833
(列)	6	0.019896	651.3155	108.5526	109	654	660	110	6	0.02016129
	7	0.220144	7206.634	1029.519	1030	7210	7196	1028	7	0.219819159
	8	0.005388	176.3816	22.0477	22	176	192	24	8	0.005865103
	32	0.426641	13966.52	436.4537	436	13952	13952	436	32	0.426197458
合計		1	32736	6150.396	6151	32728	32736	6151		1
	x	$\rho x$							x	$\rho x$
バリアブル	10	0.3125	10230	1023	1023	10230	10230	1023	10	0.3125
ノード(行)	11	0.6875	22506	2046	2046	22506	22506	2046	11	0.6875
合計		1	32736	3069	3069	32736	32736	3069		0.3125
rate		0.501007762								0.501056739

.. 行列内の「1」の総数TP=1023×32=32736

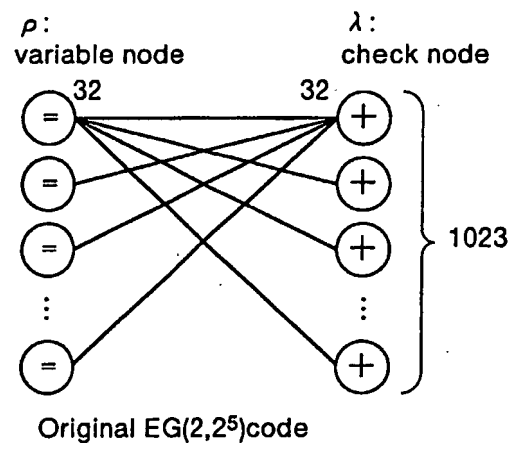
第5図

$d_i$	32		
rate	0.5		
	$x$	$\lambda_x$	No.
	2	0.178580156	2923
	3	0.149376833	1630
	6	0.02016129	110
	7	0.219819159	1028
	8	0.005865103	24
	32	0.426197458	436
	$x$	$\rho_x$	No.
	10	0.333333	1023
	11	0.666667	2046
$\sigma_{GA}$	0.962089		
$SNR_{norm}(GA)$	0.1486 dB		

第 6 図

[illegible]

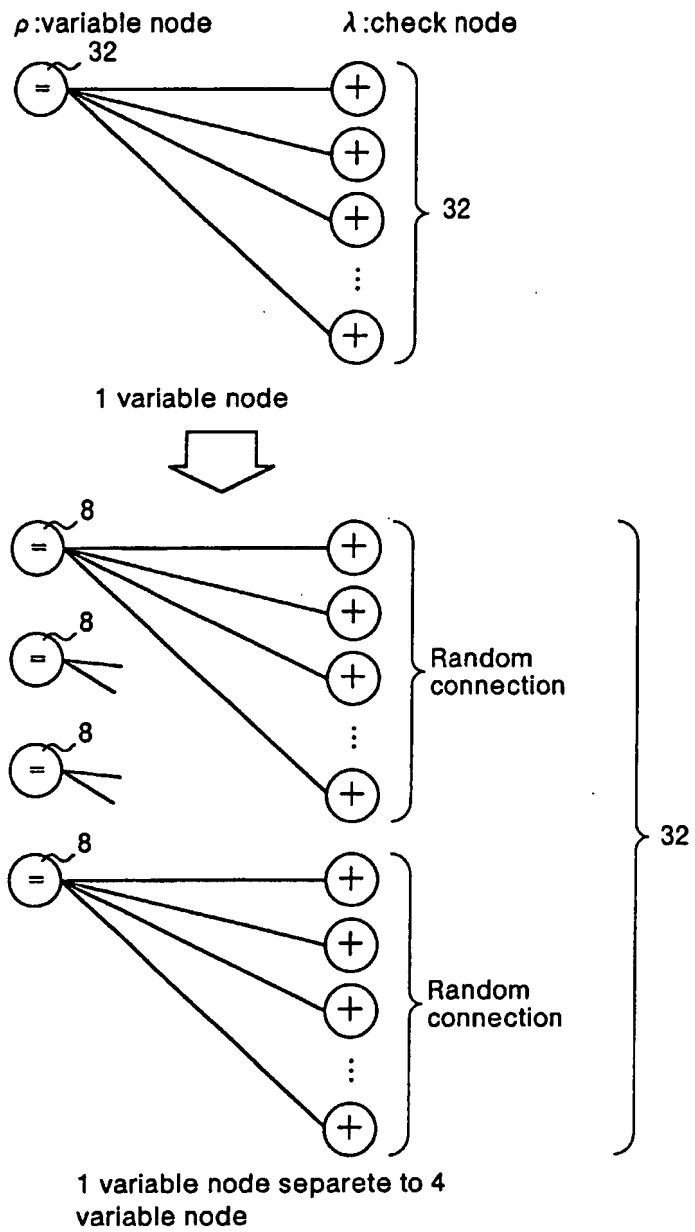
## 第7図



(=) : バリアブルノード

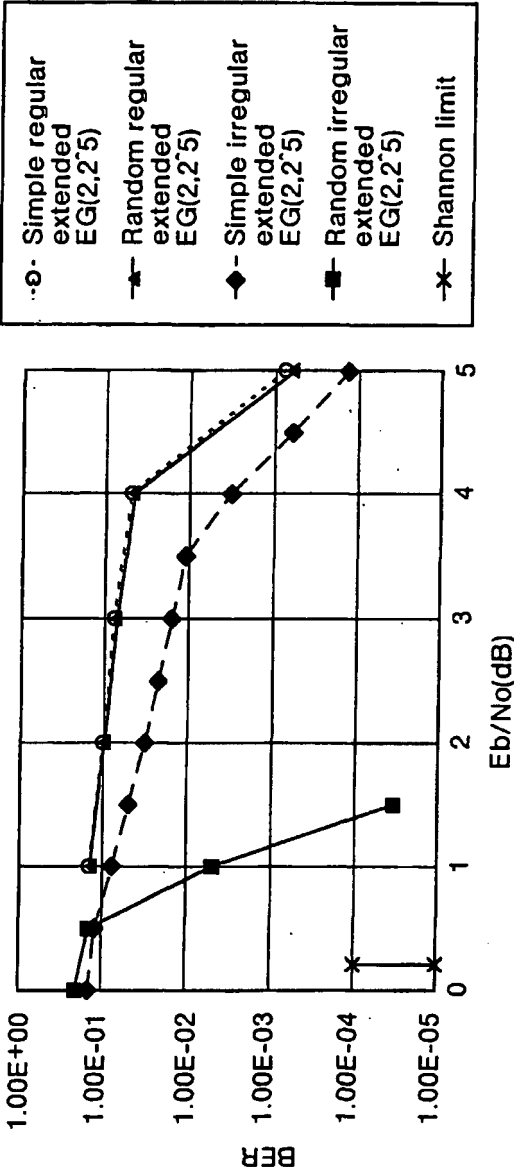
(+) : チェックノード

## 第 8 図





第9図



## 第10図

$d_t$	32		
rate	0.5		
	$x$	$\lambda_x$	No.
	16	1	2046
	$x$	$\rho_x$	No.
	32	1	1023

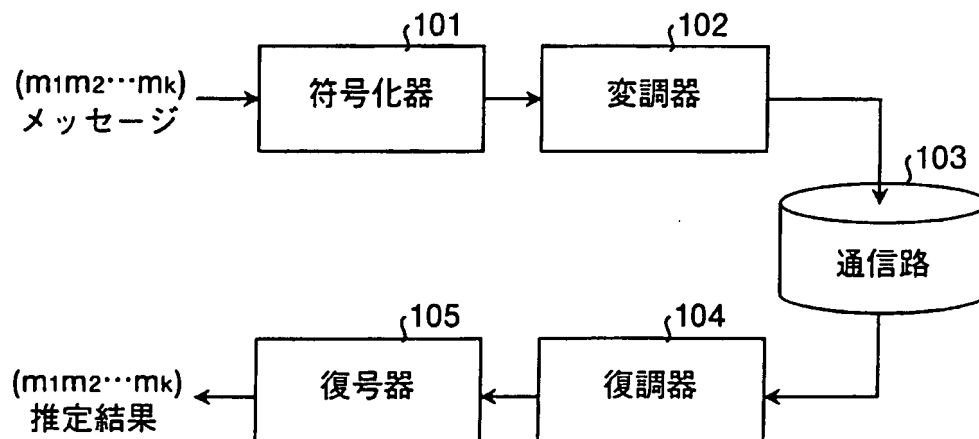
## 第11図

$d_t$	32		
rate	0.5		
	$x$	$\lambda_x$	No.
	2	0.178580156	2923
	3	0.149376833	1630
	6	0.02016129	110
	7	0.219819159	1028
	8	0.005865103	24
	32	0.426197458	436
	$x$	$\rho_x$	No.
	10	0.3125	1023
	11	0.6875	2046

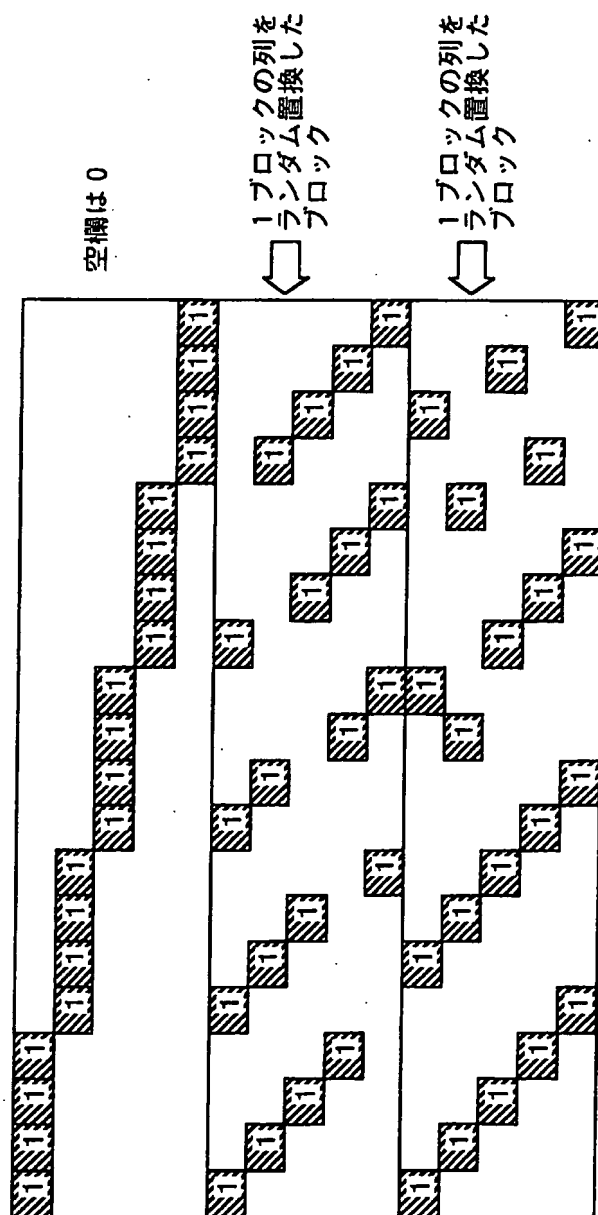
## 第12図

1	3	9	10	13	5	15	11	16	14	8	7	4	12	2	6
3	9	10	13	5	15	11	16	14	8	7	4	12	2	6	1
9	10	13	5	15	11	16	14	8	7	4	12	2	6	1	3
10	13	5	15	11	16	14	8	7	4	12	2	6	1	3	9
13	5	15	11	16	14	8	7	4	12	2	6	1	3	9	10
5	15	11	16	14	8	7	4	12	2	6	1	3	9	10	13
15	11	16	14	8	7	4	12	2	6	1	3	9	10	13	5
11	16	14	8	7	4	12	2	6	1	3	9	10	13	5	15
16	14	8	7	4	12	2	6	1	3	9	10	13	5	15	11
14	8	7	4	12	2	6	1	3	9	10	13	5	15	11	16
8	7	4	12	2	6	1	3	9	10	13	5	15	11	16	14
7	4	12	2	6	1	3	9	10	13	5	15	11	16	14	8
4	12	2	6	1	3	9	10	13	5	15	11	16	14	8	7
12	2	6	1	3	9	10	13	5	15	11	16	14	8	7	4
2	6	1	3	9	10	13	5	15	11	16	14	8	7	4	12
6	1	3	9	10	13	5	15	11	16	14	8	7	4	12	2

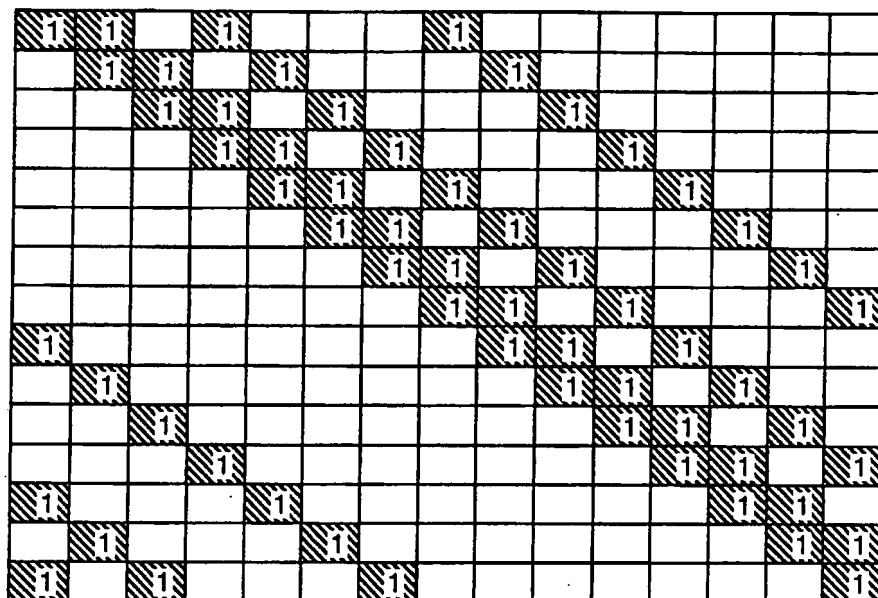
## 第13図



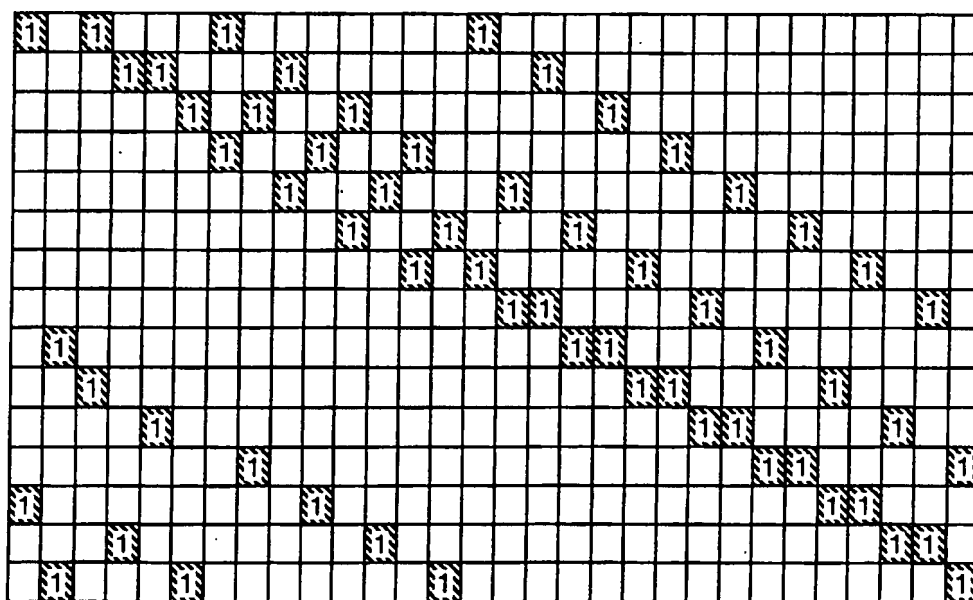
第14図



## 第15図



## 第16図



## 第17図

G0	3		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
C(1)	1	LB(1)	32	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
C(2)	2	LB(2)	2	32	7	9	11	13	15	17	19	21	23	25	27	29	31	2	4	6	8	10	12	14	16	18	20	22	24	26	28	1
C(3)	4	LB(3)	3	5	32	13	16	19	22	25	28	31	3	6	9	12	15	18	21	24	27	30	2	5	8	11	14	17	20	23	1	30
C(4)	10	LB(4)	4	7	10	32	21	25	29	2	6	10	14	18	22	26	30	3	7	11	15	19	23	27	31	4	8	12	16	1	26	29
C(5)	28	LB(5)	5	9	13	17	32	31	5	10	15	20	25	30	4	9	14	19	24	29	3	8	13	18	23	28	2	7	1	20	24	28
C(6)	20	LB(6)	6	11	16	21	26	32	12	18	24	30	5	11	17	23	29	4	10	16	22	28	3	9	15	21	27	1	12	17	22	27
C(7)	27	LB(7)	7	13	19	25	31	6	32	26	2	9	16	23	30	6	13	20	27	3	10	17	24	31	7	14	1	2	8	14	20	26
C(8)	17	LB(8)	8	15	22	29	5	12	19	32	11	19	27	4	12	20	28	5	13	21	29	6	14	22	30	1	21	28	4	11	18	25
C(9)	18	LB(9)	9	17	25	2	10	18	26	3	32	29	7	16	25	3	12	21	30	8	17	26	4	13	1	7	15	23	31	8	16	24
C(10)	21	LB(10)	10	19	28	6	15	24	2	11	20	32	18	28	7	17	27	6	16	26	5	15	25	1	22	31	9	18	27	5	14	23
C(11)	30	LB(11)	11	21	31	10	20	30	9	19	29	8	32	9	20	31	11	22	2	13	24	4	1	4	14	24	3	13	23	2	12	22
C(12)	26	LB(12)	12	23	3	14	25	5	16	27	7	18	29	32	2	14	26	7	19	31	12	1	15	26	6	17	28	8	19	30	10	21
C(13)	14	LB(13)	13	25	6	18	30	11	23	4	16	28	9	21	32	28	10	23	5	18	1	24	5	17	29	10	22	3	15	27	8	20
C(14)	9	LB(14)	14	27	9	22	4	17	30	12	25	7	20	2	15	32	25	8	22	1	31	13	26	8	21	3	16	29	11	24	6	19
C(15)	25	LB(15)	15	29	12	26	9	23	6	20	3	17	31	14	28	11	32	24	1	5	19	2	16	30	13	27	10	24	7	21	4	18
C(16)	11	LB(16)	16	31	15	30	14	29	13	28	12	27	11	28	10	25	9	1	8	23	7	22	6	21	5	20	4	19	3	18	2	17
C(17)	31	LB(17)	17	2	18	3	19	4	20	5	21	6	22	7	23	8	1	32	25	10	26	11	27	12	28	13	29	14	30	15	31	16
C(18)	29	LB(18)	18	4	21	7	24	10	27	13	30	16	2	19	5	1	24	9	32	28	14	31	17	3	20	6	23	9	26	12	29	15
C(19)	23	LB(19)	19	6	24	11	29	16	3	21	8	26	13	31	1	22	8	25	11	32	2	20	7	25	12	30	17	4	22	9	27	14
C(20)	5	LB(20)	20	8	27	15	3	22	10	29	17	5	24	1	18	5	23	10	28	15	32	9	28	16	4	23	11	30	18	6	25	13
C(21)	13	LB(21)	21	10	30	18	8	28	17	6	26	15	1	12	31	19	7	26	14	2	21	32	18	7	27	16	5	25	14	3	23	12
C(22)	6	LB(22)	22	12	2	23	13	3	24	14	4	1	4	24	13	2	22	11	31	20	9	29	32	29	19	9	30	20	10	31	21	11
C(23)	16	LB(23)	23	14	5	27	18	9	31	22	1	25	15	5	26	16	6	27	17	7	28	18	8	32	11	2	24	15	6	28	19	10
C(24)	15	LB(24)	24	16	8	31	23	15	7	1	13	4	26	17	8	30	21	12	3	25	16	7	29	20	32	26	18	10	2	25	17	9
C(25)	12	LB(25)	25	18	11	4	28	21	1	30	22	14	6	28	21	13	5	28	20	12	4	27	19	11	3	32	12	5	29	22	15	8
C(26)	3	LB(26)	26	20	14	8	2	1	14	7	31	24	17	10	3	27	20	13	6	30	23	16	9	2	26	19	32	31	25	18	13	7
C(27)	7	LB(27)	27	22	17	12	1	27	21	15	9	3	28	22	16	10	4	29	23	17	11	5	30	24	18	12	6	32	21	16	11	6
C(28)	19	LB(28)	28	24	20	1	7	2	28	23	18	13	8	3	29	24	19	14	9	4	30	25	20	15	10	5	31	26	32	13	9	5
C(29)	24	LB(29)	29	26	1	16	12	8	4	31	27	23	19	15	11	7	3	30	26	22	18	14	10	6	2	29	25	21	17	32	7	4
C(30)	8	LB(30)	30	1	23	20	17	14	11	8	5	2	30	27	24	21	18	15	12	9	6	3	31	28	25	22	19	16	13	10	32	3
C(31)	22	LB(31)	1	28	26	24	22	20	18	16	14	12	10	8	6	4	2	31	29	27	25	23	21	19	17	15	13	11	9	7	5	32
C(32)	32	LB(32)	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2

Basic Random sequence

permutation pattern of basic random sequence

## 第18図

q	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
L <sub>h</sub> (1)	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2
L <sub>h</sub> (2)	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4
L <sub>h</sub> (3)	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10
L <sub>h</sub> (4)	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28
L <sub>h</sub> (5)	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20
L <sub>h</sub> (6)	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27
L <sub>h</sub> (7)	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17
L <sub>h</sub> (8)	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18
L <sub>h</sub> (9)	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21
L <sub>h</sub> (10)	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30
L <sub>h</sub> (11)	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26
L <sub>h</sub> (12)	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14
L <sub>h</sub> (13)	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9
L <sub>h</sub> (14)	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25
L <sub>h</sub> (15)	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11
L <sub>h</sub> (16)	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31
L <sub>h</sub> (17)	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29
L <sub>h</sub> (18)	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5	23
L <sub>h</sub> (19)	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13	5
L <sub>h</sub> (20)	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6	13
L <sub>h</sub> (21)	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16	6
L <sub>h</sub> (22)	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15	16
L <sub>h</sub> (23)	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12	15
L <sub>h</sub> (24)	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3	12
L <sub>h</sub> (25)	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7	3
L <sub>h</sub> (26)	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19	7
L <sub>h</sub> (27)	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24	19
L <sub>h</sub> (28)	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8	24
L <sub>h</sub> (29)	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1	8
L <sub>h</sub> (30)	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22	1
L <sub>h</sub> (31)	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32	22
L <sub>h</sub> (32)	22	1	8	24	19	7	3	12	15	16	6	13	5	23	29	31	11	25	9	14	26	30	21	18	17	27	20	28	10	4	2	32

# INTERNATIONAL SEARCH REPORT

International application No.  
PCT/JP02/13518

<b>A. CLASSIFICATION OF SUBJECT MATTER</b> Int.Cl <sup>7</sup> H03M13/19, H04L1/00  According to International Patent Classification (IPC) or to both national classification and IPC		
<b>B. FIELDS SEARCHED</b> Minimum documentation searched (classification system followed by classification symbols) Int.Cl <sup>7</sup> H03M13/19, H04L1/00  Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched  Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)		
<b>C. DOCUMENTS CONSIDERED TO BE RELEVANT</b>		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2001-168733 A (Thomson-CSF), 22 June, 2001 (22.06.01), Full text; all drawings & EP 1093231 A1	1-7
A	EP 1258999 A2 (MITSUBISHI DENKI KABUSHIKI KAISHA), 20 November, 2002 (20.11.02), Full text; all drawings & JP 2002-353946 A & US 2003/0014717 A1	1-7
A	Sason, I., Shamaï, S., Onimproved bounds on the decoding error probability of block codes over interleaved fading channels, with applications to turbo-like codes. IEEE Transactions on Information Theory. Sep.2001, Vol.47, No.6, pages 2275 to 2299	1-7
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input type="checkbox"/> See patent family annex.		
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier document but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed		"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family
Date of the actual completion of the international search 18 March, 2003 (18.03.03)		Date of mailing of the international search report 08 April, 2003 (08.04.03)
Name and mailing address of the ISA/ Japanese Patent Office  Facsimile No.		Authorized officer  Telephone No.



# INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP02/13518

## C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Richardson, T.J., Urbanke, R.L., Efficient encoding of low-density parity-check codes. IEEE Transactions on Information Theory. Feb.2001, Vol.47, No.2, pages 638 to 656	1-7
A	Yongyi, Mao., Banihashemi, A.H., A heuristic for good low-density parity-check codes at short block lengths. IEEE International Conference on Communications, 2001.ICC2001. Jun.2001, Vol.1, pages 41 to 44	1-7
A	Chiani, M., Ventura, A., Design and performance evaluation of some high-rate irregular low-density parity-check codes. IEEE Global Telecommunications Conference, 2001.GLOBECOM'01. Nov.2001, Vol.2, pages 990 to 994	1-7

## A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int. Cl<sup>7</sup> H03M13/19  
H04L 1/00

## B. 調査を行った分野

## 調査を行った最小限資料 (国際特許分類 (IPC))

Int. Cl<sup>7</sup> H03M13/19  
H04L 1/00

最小限資料以外の資料で調査を行った分野に含まれるもの

国際調査で使用了電子データベース (データベースの名称、調査に使用した用語)

## C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
A	JP 2001-168733 A(トムソン・ソーエスエフ)2001.06.22 全文、全図 & EP 1093231 A1	1-7
A	EP 1258999 A2(MITSUBISHI DENKI KABUSHIKIKAISHA)2002.11.20 全文、全図 & JP 2002-353946 A & US 2003/0014717 A1	1-7

☒ C欄の続きにも文献が列挙されている。

☐ パテントファミリーに関する別紙を参照。

## \* 引用文献のカテゴリー

「A」特に関連のある文献ではなく、一般的技术水準を示すもの  
「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの  
「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)  
「O」口頭による開示、使用、展示等に関する文献  
「P」国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの  
「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの  
「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの  
「&」同一パテントファミリー文献

国際調査を完了した日

18.03.03

国際調査報告の発送日

08.04.03

国際調査機関の名称及びあて先

日本国特許庁 (ISA/JP)  
郵便番号 100-8915  
東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

近藤 聡

5K

8730

電話番号 03-3581-1101 内線 3555

C (続き) . 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
A	Sason, I. Shamai, S. Onimproved bounds on the decoding error probability of block codes over interleaved fading channels, with applications to turbo-like codes. IEEE Transactions on Information Theory. Sep. 2001, Vol. 47, No. 6, p. 2275-2299	1 - 7
A	Richardson, T. J. Urbanke, R. L. Efficient encoding of low-density parity-check codes. IEEE Transactions on Information Theory. Feb. 2001, Vol. 47, No. 2, p. 638-656	1 - 7
A	Yongyi, Mao. Banihashemi, A. H. A heuristic for good low-density parity-check codes at short block lengths. IEEE International Conference on Communications, 2001. ICC2001. Jun. 2001, Vol. 1, p. 41-44	1 - 7
A	Chiani, M. Ventura, A. Design and performance evaluation of some high-rate irregular low-density parity-check codes. IEEE Global Telecommunications Conference, 2001. GLOBECOM' 01. Nov. 2001, Vol. 2, p. 990-994	1 - 7